

STATUS

N E W S L E T T E R

VOLUME 4 • ISSUE 10

OCTOBER • 1985



PUNK PUMPKINS
BASIC XE 4.1
THE PRINT SHOP
MORE (of course...)

FILE ID techniques
Assy Lang

S. T. A. T. U. S. - OFFICERS

PRESIDENT

GENE RODRIGUEZ.....499-6021

VICE PRESIDENT

STAN HARRISON.....497-4369

TREASURER

J. C. PETTY.....463-6904

SECRETARY

DICK LITCHFIELD.....468-6964

PROG. COORDINATOR

BUCK MADDREY.....464-2100

LIBRARIAN

DAVID HOWELL.....468-3433

MEMBERSHIP CHAIRMAN

BILL WRIGHT.....481-0555

SYSOP

DOUG BOYNTON.....468-6242

EDITOR

GENE RODRIGUEZ III.....490-2951

=====

S. T. A. T. U. S. BBS

300/1200

804-468-1096

Editor's File

If there's one thing editors hate it's having to gush.

All editors hate it. That sinking feeling that they actually like something and now they have to tell the world about it is a threat to their image (not to mention objectivity). Well, as you may have guessed by now, I'm really excited about something: 1030 EXPRESS!

This new terminal program for the 1030 modem is/has everything... it's elegant, loaded with features (including BBS "trolling"!) it's easy to use, and fast... incredibly fast. The program was written in ACTION! (giving me added respect for that language) by Keith Ledbetter of Richmond, Virginia. Keith definitely deserves a round of applause for providing me in the telecomputing ranks with a program that makes our equipment perform at top form!

I got so excited I finally got around to formatting all my disks in 2.5... love those extra sectors...

The weather was fine and so was the turn out. I'm speaking, of course, of the STATUS picnic held at Mount Trashmore Park on the 6th. If you were there I don't have to tell you that the food was excellent and the company equally so, and if you weren't there then just ask someone who was... thanks to everyone who helped out, and a special thank you to Dave Howell, chairperson of the picnic committee for putting together such a great afternoon.

This issue serves to get us back on track after the August hiatus, with the continuation of Chris Crawford's Assembly Language course: Part II, Assembler Math... reviews by Doug Boynton: BASIC XE 4.1... and Bob Alkov: PrintShop... along with the usual assortment of random oddities... sorry about the delay Bob, and if anyone else out there thinks I may have misplaced or overlooked their Newsletter submission please let me know (I'm not too proud to apologize).

Finally, I've been thinking about making the December issue a "Best of the Year" review, I'd like your thoughts on the matter or suggestions for things that should be included, either in person or on the STATUS BBS...

Assembly Language

By Chris Crawford

LECTURE TWO 6502 ARITHMETIC

ANTIC PUBLISHING INC., (c) 1985

REPRINTED BY PERMISSION

In this lecture I will take up the problem of arithmetic on the 6502. I choose this topic only because it is fairly simple to do on the 6502. There are a couple of nerve-jangling problems associated with 6502 arithmetic, but I will breeze over those in a very cavalier fashion.

Before we can do arithmetic, though, you must know a little bit about number systems. There are three that you must know: decimal, binary, and hexadecimal. Decimal is the standard numbers you have used since grade school. You count 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, and then you reach 0 again, so you put down a 1 in the tens place and resume counting from 0.

Binary works the same way, except that there are only two digits, not ten. The two digits are 0 and 1. You count 0, then 1, then you reach 0 again, so you put down a 1 in the twos place and resume counting from 0. Thus, counting from 0 to ten in binary like this:

Decimal	Binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010

In binary, instead of having ones, tens and hundreds places, we have ones, twos, fours and eights places. It takes a lot more digits to express a number in binary, but then again, we have only the two numerals 0 and 1 to work with, so what does one expect?

The hexadecimal number system is a base-16 system. In this system, you count from 0 to 16 like so: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F,10. The 10 in hexadecimal really means 16 in decimal. So 10 is 16, right? Black is white, truth is lies... stay with assembly language long enough and you'll believe anything. Actually, it's easy to avoid confusion. We use little prefixes to tell you and the computer whether a number is expressed in decimal, binary or hexadecimal. No prefix means decimal. A \$ prefix means hexadecimal; a % means binary. Thus %10 means 2 while \$10 means 16, but 10 means just plain old 10. Hexadecimal is not hard to learn at all; if you go into any store you will see that they use hexadecimal on all their signs.

ADDITION AND SUBTRACTION

Addition with the 6502 is very simple; it uses the ADC instruction. This instruction means "ADD with Carry"; I'll get to the Carry part in just a moment. For now, let me explain the instruction. The ADC instruction has an operand, normally a location in memory. When the instruction is executed, it takes the contents of that memory location and adds that value to the value in the accumulator.

It leaves the sum of the numbers in the accumulator. This of course destroys the old value in the accumulator. you can use the immediate mode of addressing with the ADC instruction, in which case it adds the value itself. thus, "ADC #9" will add a 9 to the contents of the accumulator, while "ADC FISH" will add the contents of address FISH to the accumulator.

Subtraction is just like addition. The instruction to use is SBC, which means "Subtract Borrowing Carry". Again, I'll tell you about the Carry part in a moment. This instruction subtracts the operand from the contents of the accumulator, leaving the result in the accumulator. It also can be done in either immediate mode (e.g. SBC #5) or absolute mode (e.g., SBC GOAT).

WORD LENGTH PROBLEMS

If that were all there were to arithmetic with the 6502, programmers would be paid a lot less. The first killer problem is that the 6502 uses 8-bit words; that is, the numbers that the 6502 stores and works with are only 8 bits wide. This means that the biggest number the 6502 can comprehend is 255. Uh-oh! What happens if you want to have a checkbook balancing program and you have more than \$255? What happens if you get more than 255 points in your "Decapitate the Orphans" game? In fact, what happens if you just ignore the limit and add, say 10 to 250? Well. Believe it or not, the 6502 will give you answer of 4. Why? The number system that the 6502 uses is like a wheel, with 0 at the top, counting clockwise 1,2,3,...all the way up to 255, which lies right next to the 0. If you go up from

255 you just wrap around past the 0 and start all over. Similarly, if you subtract 2 from 0 you'll get 254.

The solution to all this is provided by the Carry bit, discussion of which I've been putting off until now. The Carry bit is a flag that the 6502 uses to remember when it has done arithmetic that carried it over the boundary between 0 and 255. By using it properly, you can solve your arithmetic problems.

The first trick to using the Carry bit is to use multi-byte words. This means that, instead of using a single byte to store a number, you use several. For example, if you use two bytes to remember a number, you can store a number as large as 65,535. Three bytes lets you go to 16,777,215. Four bytes lets you go to 4,294,967,295. Big enough for you?

To use multi-byte arithmetic, you set up a series of additions or subtractions. Suppose, for example, that you want to add two two-byte words. The program fragment to do this would look like this:

```
LDA LOFISH
CLC
ADC LOGOAT
STA LOANSR
LDA HIFISH
ADC HIGOAT
STA HIANSR
```

This little fragment of code assumes that the first two-byte value is called (LOGOAT, HIGOAT), and that the second is called LOANSR, HIANSR). The new instruction, CLC stands for "Clear Carry" and it means that the Carry bit should be set to 0. It should always be used with all additions except chained additions like this one.

The code does the following: first it adds the two low values. If the addition resulted in a wraparound(result greater than 255), then the Carry bit was set; otherwise, it was cleared. Then it performed the second addition, adding in the value of the Carry bit. (That's why we call it "Add with Carry".) Thus, if a wraparound occurred, an additional one was added into the high sum. This system insures that multi-byte addition works properly.

For subtraction, you use the SEC instruction ("Set Carry"). Otherwise, you handle subtraction the same way that you handle addition. In both addition and subtraction, though, the low bytes must be handled first, then the higher bytes in the proper order(lower to higher).

DECIMAL & SIGNED ARITHMETIC

There are two variations on standard 6502 arithmetic. Both are so rarely used that I will not treat them here. The first is decimal arithmetic using the Decimal flag. This allows you to set up an automatic decimal adjust mode. This is useful in certain types of arithmetic, primarily BCD arithmetic. If you don't know what this is, don't bother with the Decimal flag. Your program should always begin with the instruction CLD, which means "Clear Decimal Flag". I will tell you this just once: failure to clear the decimal flag is the source of the most frustrating and impossible-to-trace bug in the 6502. Every single program should start with the instruction CLD.

The second arcane bit of 6502 arithmetic is sign arithmetic. It uses the V flag ("oVerflow"). Signed arithmetic is always confusing and seldom useful. In 7 years of working with the 6502, I have never had need of it. Don't bother.

LIMITATIONS ON 6502 ARITHMETIC

There are quite a few limitations on 6502 arithmetic. There is no facility for multiplication and division; you have to write subroutines to do that. You must design your programs to make do with 8-bit words; failing in that, you must use multi-byte arithmetic, with its consequent price in speed and RAM. All in all, arithmetic is a real pain on the 6502. This is the major reason why most 6502 programs do so little arithmetic.

PRINTED BY

Professional Blueprinters Of Tidewater, Inc.
3324 Virginia Beach Boulevard
Virginia Beach, Virginia 23452
(804) 498-7069

BASIC XE 4.1

By Doug Boynton

Sure, you bought the 130XE because it was new, it was cheap, and it had 128K of memory waiting for you to fill it up. But if you were like many of us, once you got the thing in the door, you were anxious to find something you could DO with it. I mean, aside from the RAMDISK, there wasn't much that extra 64K is good for.

Until now. BASIC XE from Optimized Systems Software is the first piece of software I've seen specifically for the 130XE that takes advantage of the extra memory.

I hesitate to say FULL advantage, because there are a few things they're not telling you in the ads. Minor points, to be sure, but let's get them out of the way first.

If you're expecting to fire up your computer and see something like 90K bytes free, it doesn't quite work that way. In a nutshell, BASIC XE uses the extra memory in the 130XE for program storage (64K), while using what's left in the normal memory space for variables and the like (roughly 30K). While that's a healthy amount of memory, it's not like firing up the Commodore 128, and seeing, "122000 bytes free".

The ads also don't tell you that to do that, you have to load in a series of machine language routines from a disk.

However, BASIC XE is from the same folks that brought you ATARI 8K BASIC, BASIC A+, and BASIC XL, so you'll find it compatible with most of your existing ATARI BASIC software. Not all of it, though. BASIC XE is larger than ATARI 8K BASIC, and has squirreled away some machine language routines in places where your basic program may have its own subroutine. I've found this to be the case in several instances.

I "grew up" using Microsoft Basic, and one of the things I was looking forward to was the IF/THEN/ELSE statement ATARI BASIC sorely lacks. The advertisements certainly lead one to believe that this is included in BASIC XE. It is not. The "ELSE" referred to in the ads turns out to be an "IF/ELSE" statement similar to the "CASE" statement familiar to PASCAL users. I reread the ad. Technically, it was correct, but it was misleading, nonetheless.

Okay, enough for the critical remarks. Why would you want to shell out \$50+ for another version of BASIC? Well, it certainly adds new life to some of those tired, old BASIC games. With a revised floating-point math routine, BASIC XE is faster even without its extender disk. But one of the new commands is "FAST". Load in the m/l routines with your program, make "FAST" the first line of the program, and watch "CHICKEN" start to look like an arcade game. Or cut down drastically the "thinking" time on games like "REVERSI".

Other goodies include program listings in upper and lower case, automatic indentation of loops for easy reading, and program renumbering.

String arrays are also possible (hooray!), and there's no need to DIMension arrays if you can accept default values. Error messages are in English, not code numbers.

Is it worth it? It depends on how serious a programmer you are. I was very pleased to have an extra 30K or more to work on the BBS program with; especially since ATARI BASIC was leaving me with about 200 bytes to play with. For sloppy intermediate-level programmers like me, it's a godsend. The speed is nice to have not only for games, but for those "thinking" routines. Advanced programmers or program users who have a lot of programs with small m/l routines may be disappointed with some incompatibility problems, but may enjoy writing programs from scratch using its capabilities. Beginners would do well to learn it from the ground-up, or else master ATARI BASIC first, so you appreciate its capabilities more.

I've been playing around with the FOREM BBS program and BASIC XE. When I get the machine language routines relocated, I'll end up with about 30K for program expansion, and about 29K left for variables. That, and the program runs faster, too. Not bad.

I'm not sorry I bought it. If you think some added speed, a few new commands, and a whole bunch more memory for your programs is worth it, chances are you wouldn't be sorry, either. But don't throw ATARI BASIC into the trashcan, either.

Incidentally, I've heard some reports of problems with version 4.0 of BASIC XE. If you're an owner of that release, you may want to contact your dealer or OSS for more information. Aside from the ones mentioned above, I haven't had any serious problems with version 4.1.

STATUS Minutes

First Meeting-- September 5, 1985

President Gene Rodriguez opened the meeting by welcoming the members and visitors. Status officers were then recognized.

Buck Maddrey (Activities) introduced Doctor Hal Breedlove, O.D. Doctor Breedlove made a presentation on "VDTs and Vision". A short Q&A period followed the presentation.

Gene announced the latest information from Atari and other hardware and software manufacturers. Preliminary information on the 256K Randisk modification for the old Atari 800 by Gene Rodriguez, Randy Blain, and Dick Litchfield was released.

Under business, two motions were recieved, seconded, and passed.

1. A club picnic will be planned for the near future. Dave Howell was appointed chairman of a committee to plan the event and report back at the next meeting.

2. The club will purchase a complete set of Atari service manuals and diagnostic cartridges. These will placed into the club library.

J.C. Petty presented the treasurer's report and an update on the purchases from Best Electronics.

2nd Meeting-- 19 September 1985

President Gene Rodriguez opened the meeting by welcoming all members and visitors.

Dick Litchfield demonstrated the "Monster 800" with 288K of memory. The most useful aspect is the randisk capability. Modification instructions and software are Public Domain and are available in the club library.

Jerome Ruben of I/O Computer demonstrated the new Atari 520ST computer.

Dave Howell announced the picnic date. The picnic will be held 6 October at Mt. Trashmore from 2PM to closing.

Doug Boynton (BBS Sysop) reported on the telecomputing workshop conducted at his home on 14 September. Another workshop on CompuServe will be held in October.

J.C. Petty presented the treasurer's report and the meeting was adjourned.

Dick Litchfield##
Secretary

800XL BASIC Disabler

MILE HIGH ATARI Newsletter 8/85

This article is for those of you who load binary files with an AUTORUN. SYS menu loader. This will explain how to boot them without holding down the option button.

The XL series enables and disables basic at address hex \$D301, decimal 54017. The 400/800 series uses this address for input from joyports #3 and #4. The XL series uses this for a variety of different things. For example, bit 0 at hex \$D301 says the operating rom is active or you are using ram below it. Bit 1 says BASIC is enabled or disabled. One of the other bits at this address tells if the diagnostic rom is enabled or disabled. The normal values for \$D301 are \$FF (basic disabled) and \$FD (basic enabled). All we have to do is add some instructions to the loader program to access ram not basic, thus ridding you of holding down the OPTION button to load binary files. If you're into machine code you might try this. LDA #\$FF, STA \$D301, or since ATARI files can have multiple segments (each having their own start and run address) and all files start with two \$FF bytes, you could specify that a file starts at \$D301 and ends at \$D301, and consists of only one byte \$FF. This would put an \$FF byte at \$D301, enabling ram and disabling BASIC. However, there is an easier way (of course !!).

FOR ATARI DOS 2.05

1. Boot DOS while holding the OPTION button (last time you'll need to do this).
2. Place AUTORUN.SYS file in drive
3. Hit the 'E' option to rename the loader. Ex: AUTORUN.SYS, AUTORUN.OLD.
4. Hit the 'K' option to binary save AUTORUN.SYS at starting address of D301 and ending address of D301. Ex: AUTORUN. SYS, D301, D301
5. Last, hit the 'C' option to copy a file. This will be used to append AUTORUN.OLD to the just saved AUTORUN.SYS file. Ex: AUTORUN.OLD, AUTORUN.SYS/A

FOR OSA+ or DOS XL

1. Boot DOS while holding OPTION. For the DOS XL menu hit 'Q'
2. Put AUTORUN.SYS file in drive.
3. Type: RENAME AUTORUN.SYS AUTORUN.OLD
4. Type: SAVE AUTORUN.SYS D301,D301
5. Type: COPY -AF AUTORUN.OLD AUTORUN.SYS

You now have an AUTORUN.SYS that will load any binary file without the pesky OPTION button. [excerpt's from Atari RAY AREA Computer Users. OCT 84]



BEST ELECTRONICS

ATARI OWNERS' BEST FRIEND

9 VOLT AC 50 WATT PWR ADAPTRS.....\$12.00
ATARI ORIG WORD PROC.....\$20.00
" ASSEMBLER CART W/ MANUAL.....\$15.00
USERS GROUP PRICES AVAILABLE!!!!

EPSON MX/FX 80 RIBBONS.....\$6.25
1020 COLOR PLOTTR BLK PENS.....\$2.90/SET

ATARI CX85 NUMERICAL KEYPAD.....\$15.00

ASK ABOUT OUR NEW LOW PRICES ON MPP/SUPRA
PRODUCTS (INCLUDING MPP MicroNet!!)

BEST ELECTRONICS
2021 The Alameda Suite 290
San Jose, CA 95126
(408) 243-6950

The Print Shop

Reviewed by Bob Alkov

Here at last is a program that is not only fun for the whole family, but is so simple to use that it needs no documentation! The manual states on the overleaf that "The Print Shop is so easy to use, you don't need to read this manual. Just start up the program and follow the prompts. You'll be printing in minutes!" And they're right! The first time you use the system you are prompted to tell the program which printer you will be using. Everything that follows is menu driven and leads you by the hand one step at a time.

The Print Shop allows you to print your own personalized greeting cards for any occasion. You can also make signs and banners, create letterheads, and design your own graphics. There is even a Screen Magic option that allows you to view a variety of kaleidoscopic images automatically generated by the program. Then you can freeze the action, print a message on it and dump the whole thing into the printer for a hard copy.

There are eight different type fonts to select from and 50 symbols plus 10 patterns to choose. Nine border designs are available to dress up your greeting cards and posters. You can create your own distinctive logo for a letterhead by using the Screen Editor. You can put your name and address with your logo at the top of the page, at the bottom or in both locations. You are provided with a sample of colored stationary paper with edge perforations to fit in your tractor feed and envelopes. You can run it through printing your letterhead, then back through printing using a word processor.

The kids can have hours of enjoyment creating banners for parties or school activities. They can make greeting cards for family and friends, invitations to parties and signs for their rooms. "Keep Out-Genius at Work" marked with skull and crossbones, for example. Symbols can be colored in using crayons, felt tip pens or colored pencils. You can even print in several colors by changing the ribbon for colored ones and feeding the paper back through the printer for borders, logos, lettering, etc.

You can use a joystick, Koala pad or Atari touch tablet for creating graphics, but I prefer using the keyboard cursors for editing to put in the fine touches pixel by pixel. You can save your creations to a blank disk or change existing designs in the program with the Screen Editor and save them to the disk to be recalled later.

The Print Shop comes on a disk and requires 48K of memory, a printer and an interface. Refer to the package for a list of printers and interfaces you can use with the program. I was worried because they only listed the B50, Ape Face, and Microbits 1150/Microprint interfaces. I had an MPP 1100, but it worked fine. Listed at \$45.00, I got my Print Shop on sale for \$36.00. It was well worth the price. I rate it *****.

Review

BY SCOTT NATHENS

ATARI GAMES AND RECREATIONS

by Herb Kohl, Ted Kahn, Len Lindsay, & Pat Cleland
Reston Publishing Company, 1982

Atari Games and Recreations offers a variety of BASIC programs for the beginning and advanced hacker. The book starts with very easy games which build upon fundamental programming ideas and quickly leads to more sophisticated game concepts. If you are a BASIC hacker and type a lot of programs from magazines, Atari Games and Recreations is an excellent learning resource. Its main intent is to help you learn programming styles and encourage you to develop your own game ideas. The book enables you to play as you learn and ultimately take power over your Atari.

Atari Games and Recreations is divided into three parts -- learning about the Atari through games, games for your Atari, and graphics and sound. The table of contents includes elementary name, word, and race games in Part 1. Part 2 develops the programming skills of Part 1 by presenting elaborations on games. Part 3 is entitled "The Atari Special" and introduces graphics, sounds, and color techniques to the BASIC programmer.

One of the major highlights of this book is the comprehensive appendices. A clearly explained error message section helps make it a breeze to debug program problems. Appendices on timing input and a scorekeeper subroutine enhance game designing. Additionally, a quick reference guide to BASIC commands for the Atari is incorporated in AG&R's appendix.

AG&R can serve as an exciting learning tool for beginners as well as advanced programmers. It demonstrates how games form a beautiful domain for understanding Atari BASIC programming skills. Furthermore, it encourages computer

literacy via actual play. If you're a beginner or advanced hacker or just spend a lot of time typing programs, then A6&R promises to be both educational and entertaining.

A6&R is available through the Virginia Beach Public Library System -- Call number 794/k79.

Scott Mathews

Telecomputing

By Doug Boynton

"The file I downloaded from the BBS won't work!"

That's the number-one question I get from users. This time, we're going to take a look at some of the reasons... answers... and how you can tell what kind of file you're dealing with.

The first thing you need to know -- is what kind of file you're downloading. There are several types of files available on the STATUS BBS (and other systems):

LISTed Basic: A Basic file in LISTed form. You don't see this type of file too often on BBSsystems much anymore, except as patches (fixes, modifications) to SAVED basic files. This is a file that has been stored using ATARI BASIC'S <LIST "D:FILENAME.EXT"> command. If it is a patch to a SAVED Basic file, you would want to <LOAD> the program to be modified, then <ENTER> the modification, then <SAVE> resulting modified program. In its simplest form, a LISTed program is <ENTER>ed from Basic's "READY" prompt.

SAVED Basic: A Basic file in tokenized form. The most common way of storing an ATARI BASIC program. This is a file that has been stored using the <SAVE "D:FILENAME.EXT"> command. This type of program is either <LOAD>ed and then <RUN>; or, more simply, <RUN>.

BINARY: This is usually a machine-language file. The usual way to use a file like this is to <L>oad it from DOS. Many Binary files can also be made as LOAD-AND-GO files by renaming them as AUTORUN.SYS (or your DOS's version of that filename).

BINARY DATA: This might be a KOALA PICTURE file... an AMS MUSIC file... or something else entirely. This type of file is usually information that is used by a program... for that program to work. This type of file is not a program in itself -- and won't run as a program. For example, this is the type of file that results if you save a KOALA PAD picture... or save the results of a session with SYNFILE or SYNCALC.

TEXT (ALSO ASCII TEXT): This is usually a documentation file. It often contains "DOC" or "TXT" somewhere in its name. This is not a program. It is a file that is meant for you to read. The simplest way to do this is to <C>copy the file to the screen from DOS. When DOS prompts you for where you want to copy from, and to, respond with, <D:FILENAME.EX", E:>, and toggle the file scrolling up the screen with <CTRL-I>.

Now, you say, that's easy for me to say, but what if you don't know what kind of file you're dealing with? What if the only reason you downloaded the file in the first place is that it had an interesting-sounding name?

The first place to turn for information on the file is the Index. On the STATUS BBS, the Index is located in the (<*) Database, choice #3. This will give you information on what the file is, what it does, and what type of file it is. You should always check the Index when downloading a file. This also gives you information on files that come as a set. For example, files that need other files to run... won't run by themselves. (See Binary Data, above)

Another way to identify a file is by <C>copying it to the screen from DOS. This is a bit tricky, but here's how you do it. Use the steps outlined above for <C>copying the file to the screen. As soon as that begins, halt the output using <CTRL-I>. Look at the first line or two printed out. If the first three characters printed are hearts, this is a SAVED Basic file. If the first two characters are inverse arrows, this is a BINARY file. If the first line is a short line of junk characters, followed by a blank line, followed by more junk, chances are this is a KOALA PAD Binary Data file. If the first couple of lines are numbers... one number to a line... chances are this is an AMS MUSIC file. The only way to get good at identifying files this way is to copy files to the screen that you can identify. You'll begin to pick up the patterns.

One of the Biggest mistakes Atari made was in dropping the BASIC manuals from the XL line of computers... there was a wealth of information in there about file structures. Get your hands on an old BASIC or DOS manual. Read up on how files are stored... try to wade through the technical stuff. You'll get a big insight on what different types of files look like when you <C>copy them to the screen from DOS.

Okay, you say, I know all this, and the stupid file still won't run. What now? Here's the process I follow when I'm trying to work with a file that won't work:

1) Switch to ATARI DOS 2.0. Sure, that "ULTRASUPERSMART DOS 5.0" says it's compatible with ATARI DOS 2.0... but few of them really are. Go back to basics (not the language) here... and give it a shot. Many binary files are not transportable from one DOS to another, because the programmers tend to squirrel away data in out-of-the way places that might be used by one DOS or another.

2) If you're trying to <L>oad a Binary file, try it without BASIC. For the newer XL/XE computers, this means booting the system while holding down the <OPTION> key. For the older models, it means taking the BASIC cartridge out.

3) If you're using a XL or XE, try booting the system using one of the translator disks floating around. If that doesn't work, try another translator. (There are several -- all available through the STATUS library.)

If you've done all these things, and the program STILL doesn't work, try downloading it again, and start over. If it STILL doesn't work, ask the SYSOP to check it out -- or post a message to the person who uploaded it, asking if there's anything special about the file. You might also post a message to others... to see if they have had problems.

I hope I've given you a bit more than the "You get what you pay for" answer that many SYSOPS (including me) use from time to time. You deserve more than that.

NEXT TIME: COMPUTER MORALITY....



President's Column

The First Annual STATUS Picnic held on 6 October was well received by the members and their families, we are looking forward to next years...

I would like to thank Dave Howell and his picnic committee for an outstanding feed, and Stan Harrison for those soft drinks...

I would like to congratulate Trent Ducley of the Starfleet Atari User Group for being winner of our Public Domain Programmer of the Month Award. This award is a form of encouragement for the public domain programmer, and was based on the acceptance of his program by a large majority of our members... Keep up the good work Trent... hope to hear from you soon.

Halloween is just around the corner and will it be a trick or a treat from Atari.... last month Atari announced via Compuserve that Atari would be sending dealers and user groups that participated in the initial " 520ST rollout program" copies of PD programs for the ST that include: a drawing program called DOODLE, NEOCHROME a paint program, and ST Writer a word processor a la Atariwriter with a utility for uploading 8-bit Atariwriter files to ST. We did not participate so we will have to wait until our group buys an ST. I have been giving out information that I have been gleaning from our exchange newsletters to (non-members) folks that have bought the ST. And by some strange luck these folks are going to other areas, so I have been referring them to other user groups. So, If someone shows up at your meeting, and say's STATUS sent me... Welcome them, after all we are all ATARIANS.

For you future ST owners a Newsletter called SIXTEEN/THIRTY-TWO APPLICATIONS has gone to the printers, the current issue of 'S/T' for September is available at \$5.00 an issue or a 6 issue subscription for \$25.00, for Ten bucks they will send you the current issue on disk (disk version 6 issues for \$50.00)... Send your name and address with your remittance to:

'S/T' Applications
10760 Hwy 116
Forestville, CA 95436.

Atari renaissance... HOME COMPUTER magazine has added ATARI to it's line up; could this be an indication of things to come?... Remember back a couple of years ago when ATARI was being dropped for lack of interest(\$\$\$)? I'm glad to see this trend reverse itself... Congratulations HOME COMPUTER MAGAZINE...

S.T.A.T.U.S.

Opinions expressed in this publication are those of the individual authors and do not necessarily express or reflect the opinions of the Southside Tidewater Atari Technical Users Society. Some of the material contained herein may have been taken from bulletin boards and newsletters of other groups and should not necessarily be construed as fact.

The material herein may be copied freely provided that full credit is given to the original author and the Southside Tidewater Atari Technical Users Society. S.T.A.T.U.S. is not affiliated with Atari Inc.

Please address all newsletter correspondence to:

S.T.A.T.U.S.
Newsletter Exchange
4836 Honeygrove Rd.
Va. Bch., VA 23455
(804)499-6021

MEETINGS: S.T.A.T.U.S. meetings are held on the first and third Thursdays of each month at the Baylake Pines Private School, 2204 Treasure Island Dr., in Virginia Beach. at 6:30 p.m. All interested parties are welcome to attend.

NEWSLETTER ARTICLES:

Submitted articles are preferred on disk text files, but will be greatfully accepted on hard copy (including handwriting) if you do not have a disc drive. If you have a modem, you can upload your articles to the Editor by calling 499-6021. Articles may be submitted anytime, but will probably not make that month's newsletter if submitted less than one week before the regular meeting date.

=====

S.T.A.T.U.S. BBS
804-468-1096

S.T.A.T.U.S
Newsletter Exchange
C/O Gene Rodriguez
4836 HONEYBROVE RD.
Virginia Beach, VA 23455

(First Class Mail)

Atari Cost, Services

Must send in: {

800x1 receipt for Rev. C.
Dos 3 unpunched, w/in 90 days
or buying drive